PN - DE19742906 A 19990506
PD - 1999-05-06
PR - DE19971042906 19970929
OPD - 1997-09-29
TI - Optimizing products and their production processes by neural analysis
AB - Product and/or process optimization is carried out by neural analysis based on self-organizing maps (SOMs). A process for optimization problem solving in research, development and construction and for optimization of technical or chemical products and their production processes comprises collecting and processing all the values relevant to the optimization by a neural analysis based on self-organizing maps (SOMs), in which a topology producing, non-linear projection of data of the relevant parameter and the associated target values is performed on a multi-dimensional SOM. The target value is optimized by displaying as a SOM component map in either height coded or color coded form and, after selecting values for the target value, the underlying parameter combinations are calculated and output.
IN - OTTE VIKTOR PROF DR ING (DE); OTTE RALF DIPL ING (DE)
PA - ABB PATENT GMBH (DE)
EC - B01J19/00B ; G05B13/02C1
IC - G06F17/50 ; G05B19/4097 ; B62D65/00 ; C07B61/00 ; C12N15/00 ; B01J19/00 ; F01K13/02
CT - DE19637651 A1 [ ]; DE4443192 A1 [ ]; DE4029774 A1 [ ];
DE3710990 A1 [ ]
CTNP - [ ] WENZEL,Lothar: Optimalstrategie. In: mc extra, 10/95, S.30-33;
- [ ] HAMPEL,Rainer, CHAKER,Nasredin: Dampfturbinen- regelung mit Fuzzy-Logik. In: atp - Automatisierungstechnische Praxis 37, 1995, 6, S.32-41;
- [ ] GEHLEN,Stefan, u.a.: Einsatz Neuronaler Netze zur Kontrolle komplexer industrieller Prozesse. In: at - Automatisierungstechnik 43, 1995, 2, S.85-91;
- [ ] GIEREND,Chr.: Feuerungsregelung durch Fuzzy Control. In: VGB Kraftwerkstechnik 76, 1996, H.6, S.487-497;
- [ ] Prozessmodelle auf Basis Neuronaler Netze. In: atp 4/97, S.66;
- [ ] SPUR,Günter, KRAUSE,Frank-Lothar: CAD-Technik, Carl Hanser Verlag, München, Wien, 1984, S.282- S.285;
- [ ] ROJAS,PaUl: Theorie der neuronalen Netze, Springer-Verlag, Berlin u.a., 1993, S.339-359

TI - Optimizing products and their production processes by neural analysis
PR - DE19971042906 19970929
PN - DE19742906 A1 19990506 DW199924 G06F17/50 013pp
PA - (ALLM) ABB PATENT GMBH
IC - B01J19/00 ;B62D65/00 ;C07B61/00 ;C12N15/00 ;F01K13/02 ;G05B19/4097 ;G06F17/50
IN - OTTE R; OTTE V
AB - DE19742906 NOVELTY - Product and/or process optimization is carried out by neural analysis based on self-organizing maps (SOMs).

- DETAILED DESCRIPTION - A process for optimization problem solving in research, development and construction and for optimization of technical or chemical products and their production processes comprises collecting and processing all the values relevant to the optimization by a neural analysis based on self-organizing maps (SOMs), in which a topology producing, non-linear projection of data of the relevant parameter and the associated target values is performed on a multi-dimensional SOM. The target value is optimized by displaying as a SOM component map in either height coded or color coded form and, after selecting values for the target value, the underlying parameter combinations are calculated and output.
- USE - For process optimization in vehicle and machine construction, in precision engineering, for development of genetic products and power plants and components.
- ADVANTAGE - The process provides a data-based universal optimization process which permits simultaneous and continuous processing and display of all relevant optimization parameters and target values of technical or chemical products and their production processes and which provides a graphic display of the quality functions of the optimization.
- DESCRIPTION OF DRAWING(S) - The drawing shows the essential components of a system for carrying out the process.
- (Dwg.1/6)
OPD - 1997-09-29
AN - 1999-278422 [24]

(19) BUNDESREPUBLIK DEUTSCHLAND

DEUTSCHES PATENT- UND MARKENAMT

(10) DE 197 42 906 A 1

(21) Aktenzeichen: 197 42 906.8
(22) Anmeldetag: 29. 9. 97
(43) Offenlegungstag: 6. 5. 99

(51) Int. Cl.[6]:
G 06 F 17/50
G 05 B 19/4097
B 62 D 65/00
C 07 B 61/00
C 12 N 15/00
B 01 J 19/00
F 01 K 13/02

DE 197 42 906 A 1

(71) Anmelder:
ABB Patent GmbH, 68309 Mannheim, DE

(72) Erfinder:
Otte, Viktor, Prof. Dr.-Ing.habil., 42657 Solingen, DE;
Otte, Ralf, Dipl.-Ing., 69469 Weinheim, DE

(56) Entgegenhaltungen:

DE 196 37 651 A1
DE 44 43 192 A1
DE 40 29 774 A1
DE 37 10 990 A1

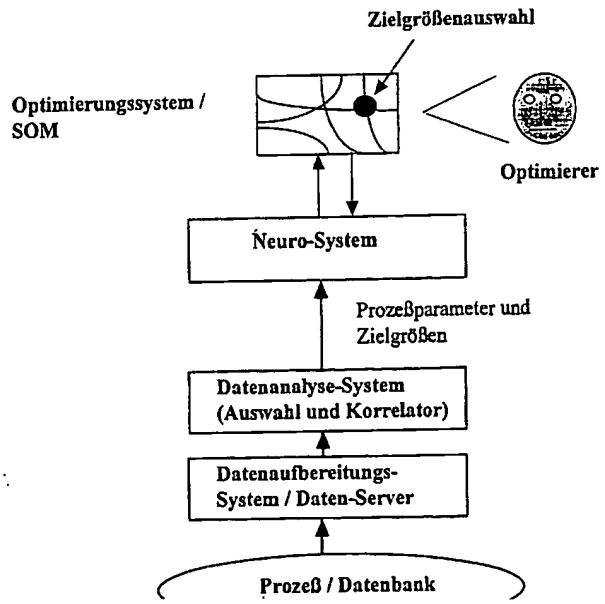
WENZEL,Lothar: Optimalstrategie. In: mc extra, 10/95, S.30-33;
HAMPEL,Rainer, CHAKER,Nasredin: Dampfturbinen-regelung mit Fuzzy-Logik. In: atp - Automatisierungstechnische Praxis 37, 1995, 6, S.32-41;
GEHLEN,Stefan, u.a.: Einsatz Neuronaler Netze zur Kontrolle komplexer industrieller Prozesse. In: at - Automatisierungstechnik 43, 1995, 2, S.85-91;
GIEREND,Chr.: Feuerungsregelung durch Fuzzy Control. In: VGB Kraftwerkstechnik 76, 1996, H.6, S.487-497;
Prozeßmodelle auf Basis Neuronaler Netze. In: atp 4/97, S.66;
SPUR,Günter, KRAUSE,Frank-Lothar: CAD-Technik, Carl Hanser Verlag, München, Wien, 1984, S.282-S.285;
ROJAS,Paúl: Theorie der neuronalen Netze, Springer-Verlag, Berlin u.a., 1993, S.339-359;
BRAUSE,Rüdiger: Neuronale Netze, B.G.Teubner, Stuttgart, 1995, 2.Aufl., S.152-164;
Prospekt: IDL Basics, RSI Research Systems, Inc., Nov. 92, S.13-16,30,31,46-55;
KOHONEN,Teuvo, et.al.: Engineering Applications of the Self-Organizing Map. In: Proceedings Of The IEEE, Vol.84, No.10, Oct. 1996, S.1358-1382;
HALICI,Ugur, ONGUN,Güçlü: Fingerprint Classification Through Self-Organizing Feature Maps Modified to Treat Uncertainties. In: Proceedings Of the IEEE, Vol.84, No.10, Oct. 1996, S.1497-1512;
GRAEDEL,T.E., McGILL,R.: Graphical Presentation of Results from Scinetific Computer Models. In: SCIENCE, Vol.215, 5. March 1982, S.1191-1198;
BAUER, H.-U., u.a.: Selbstorganisierende Karten, in: Spektrum der Wissenschaften, April 1996, S. 38-47;

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

Prüfungsantrag gem. § 44 PatG ist gestellt

(54) Verfahren zum Optimieren von Produkten und Produktionsprozessen

(57) Die Erfindung bezieht sich auf ein Verfahren zur Lösung allgemeiner Optimierungsaufgaben in Forschung, Entwicklung und Konstruktion und zum Optimieren von technischen oder chemischen Produkten und von Prozessen zur Herstellung dieser Produkte. Um eine gleichzeitige und zusammenhängende Bewertung relevanter Parameter und der dazugehörigen Zielgrößen zu ermöglichen, wird vorgeschlagen, die Einflüsse der Prozeßgrößen durch eine neuronale Analyse auf der Grundlage selbstorganisierender Karten in Beziehung zueinander auszuwerten, indem eine topologieerhaltende, nichtlineare Projektion von Daten der Versuchsparameter auf eine 2-dimensionale neuronale Karte (sogenannter SOM) realisiert wird. Nach einer Auswahl von Werten der Zielgröße (Z1, Z2) auf der SOM-Komponentenkarte werden die zugrundeliegenden Parameterkombinationen berechnet und ausgegeben.



DE 197 42 906 A 1

Description

The invention relates to a method for solving general optimisation tasks in research, development and design and for the optimisation of technical or chemical products and of processes for manufacturing these products.

The method is suitable for optimising design processes in mechanical and automotive engineering, in precision engineering or in development processes for genetically engineered products and for the optimisation of systems or parts of systems of a power station.

It is generally known that numerical methods for optimising parameters are used for optimisation problems. Here, the starting point is a set N of independent variables p_1, p_2, ... ,p_N, which are combined in a so-called parameter vector p:

$$p = (p_1, p_2, \ldots, p_N) \qquad (1)$$

The quality of such a parameter vector is described by a quality function Q. Q(p) therefore gives an indication of how well the chosen parameter vector p solves the optimisation problem. In general, Q is defined in such a way that the optimisation problem is better solved the larger or smaller Q is. If it is assumed that a minimum Q represents the optimum, then a search is made for

$$Q(p) \rightarrow \text{Minimum} \qquad (2)$$

Several methods of solution are known for this problem, which can be roughly divided into deterministic and random processes. Numerous deterministic methods, e.g. gradient methods and Newton methods, have the disadvantage in practice that partial derivations of the quality function must be available for their solution. Other deterministic methods, e.g. direct search methods, often become bogged down in the local extremes of their environment and do not find the desired global optimum, see J, Kahlert, "Fuzzy Control für Ingenieure", Vieweg Verlag, 1995, Pages 155 to 170. A strength of the classical optimisation methods - when they are applicable - consists in that they are able to find the optimum without having to have information on all areas. On the other hand, they lack universal application, as mathematical models are assumed.

Further, it is known that, particularly in research, development and design, there are optimisation tasks, which cannot be solved by specifying a single quality criterion, but which consist of target functions, which have to be optimised simultaneously, and which often place contradictory requirements on the process. This results in a so-called vectorial quality function and the target conflict is dealt with formally by polyoptimisation methods, in the results of which a search is made for the best compromise solution. As a result, the above-mentioned problems of the existing optimisation methods are again intensified so that they cannot be used in many practical applications.

Further, it is known that randomly controlled optimisation methods, e.g. Monte Carlo methods or evolution

strategies, are able to overcome the above-mentioned disadvantages; these have therefore been integrated into numerous practical applications. They assume that the scalar or vectorial quality function in the parameter space P constitutes a multi-dimensional mountain range, the minima or maxima respectively of which are to be found, see E. Schöneburg et al "Genetische Algorithmen und Evolutionsstrategien", Addison Wesley Verlag, 1994, Pages 102 to 107 and Pages 141 to 215.

A disadvantage of this randomly controlled optimisation method is that one still has no or only very little information about the make-up and the topology of the quality function Q in the state space P of the optimisation problem. In system theory, the n-dimensional vector space, in which the items of data to be investigated can be plotted above one another, is described as the state space P of a process. For example, a data example p corresponds to a point in this state space. In the following, an input space or state space P is understood to mean exactly this state space.

Furthermore, it is not know whether the optimum found is local or global. A further disadvantage lies in the correct parameterisation of the optimisation method itself, as, for example, the initially simple concepts of evolution strategies or genetic algorithms have often become too complex for application in the technical or chemical process due to adaptive step width control, different crossing-over strategies or selection variations (see E. Schöneberg et al.).

The invention is based on the object of specifying a universal data-based optimisation method, which enables a simultaneous and consistent evaluation and visualisation of all relevant optimisation parameters and target quantities of a technical or chemical product and of the process for the manufacture of this product, and which can clearly display the quality functions of the optimisation graphically.

This problem is solved by a method specified in Claim 1 for the optimisation of technical or chemical products and the optimisation of processes in research, development, design and production of these products. Here, all parameters relevant to a process are brought together and evaluated in relation to one another by a neural analysis based on self-organising maps (so-called SOMs), in which a topology-maintaining, non-linear projection of data, i.e. of all parameters and target quantities, of the relevant process is implemented on a multi-dimensional, height-coded or colour-coded neural map (SOM). As the target quantities can be plotted on this structured SOM in the form of so-called component maps with height coding or colour coding, information is obtained about the distribution of the target quantities in the whole of the parameter space P. The values of the target quantities that are sought, i.e. the minima, maxima and also particular ranges, can be selected and the associated parameter combinations determined by this means.

The only prerequisite for the method is the existence of example data for the optimisation problem. The

method thus constitutes an extension of existing optimisation methods.

The quality function for the optimisation can be a target quantity itself or can be calculated from one or more target quantities. Starting from the graphical representation of the general high-dimensional quality mountains, information is obtained about the topology of the quality functions, and the global minima and maxima can easily be read from the mountains and the associated parameter combinations can be output. Further, it must be possible to superimpose individual scalar target quantity functions, i.e. individual target quantity mountains, on one another so that a graphically orientated polyoptimisation can be carried out.

Advantageous embodiments of the method are given in further claims. Due to the holistic approach, not only the values of individual parameters but also their mutual and non-linear effects on one another are taken into account.

In the further description, a process is understood to mean any operation that has the objective of manufacturing, improving or optimising something. A process can therefore be a technical or chemical process itself and also a process in research, development and design for the manufacture of a product or for the optimisation of product characteristics. Process quantities are the quantities, which describe the process and are relevant to the optimisation task, e.g. design parameters, product characteristics or physical measurements.

A process in the above sense is described by the following vector pvec of process quantities:

$$pvec = (p_1, p_2, \ldots, p_i, \ldots, p_N, z_1, z_2, \ldots, z_j, \ldots, z_L) \quad (3)$$

where p_i is the i-th influencing quantity or the i-th process parameter and z_j is the j-th target quantity of the process, i.e. the process quantities consist of parameters and target quantities. The number $n = N + L$ gives the total number of underlying quantities for the process. N is the number of influencing quantities, L is the number of target quantities; n is therefore the number of all relevant product or process quantities. The data vector is understood to be a concrete value of $pvec_{ti}$, which is measured at a point in time t_i. The SOM method is applied based on existing data vectors from this process.

In system theory, the n-dimensional vector space, in which the items of process data can be plotted with an assignment to one another, is described as the state space R of a process. If R is arranged so that an axis in R is assigned to each quantity, a point in R is obtained for each data example $pvec_{ti}$. Such a coordinate system R is described as the state space of the process and each process state $pvec_{ti}$ at the time t_i is marked as a point in R. In the following, an input space is understood to mean exactly this state space. The output space is the 2-dimensional space, which is reproduced on the SOM map. A point in R therefore corresponds to each vector $pvec_{ti}$; a series of vectors represents a point cloud in R. The SOM method is applied based on

numerous data for the optimisation problem and thus on an above-mentioned point cloud.

In neural theory, a self-organising map (SOM) is understood to mean a "self-organising neural network", in which all neurons are arranged adjacent to one another. The self-organising neural network is a term introduced for a special class of neural networks, which structure themselves based on input signals, cf. A. Zell "Simulation Neuronaler Netze", Addison Wesley Verlag, 1994, Pages 179 to 187. In contrast with conventional neural networks, the spatial position of the individual neurons and their neighbourhood relationships play an important role in the SOM. The terms SOM and map are used with equal standing in the following.

So-called topology-maintaining representations can be realised with the help of SOMs. In this context, topology-maintaining means that the points (data points), which lie close together in the input space, will also lie close together in the output space and thus on the map. Therefore, in principle, the map represents a topology-maintaining, 2-dimensional window in the n-dimensional state space of the process.

In the method according to the invention, following an appropriate data pre-processing of the process quantities, the values of the n quantities relevant to the optimisation task are provided in a learning phase to a self-organising network. The number of available data examples should not be less than a minimum number (MA). This number is:

$$MA = 30 \cdot n \qquad (4)$$

The method also works with fewer data examples, but this causes a degradation in the results. The value of n can lie between 2 and several hundred. Process quantity or measured quantity is understood to mean both the influencing parameters with the number N and the target quantities of the process with the number L. The number of target quantities should, however be considerably smaller than the number of influencing quantities:

$$L << N. \qquad (5)$$

The learning phase takes place in two steps: first, the map unfolds in the state space of the process and then the data examples are visualised by the application of a mathematical process.

Before the learning method is used, the process quantities are pre-conditioned. The individual quantities are first scaled to a predefined value and then the noise is removed, for example. Removal of the noise allows typified optimisation examples to be multiplied and further processed, as individual items of example data, which are always subject to a systematic and statistical error, are generalised in this way. In doing so, the noise content must not be so large, however, that it qualitatively changes the existing data distributions.

The unfolding of the self-organising map in the state space of the process is realised by means of algorithms

based on a neural algorithm according to T. Kohonen. The "self organising neural SOM algorithm" was introduced by Kohonen in 1982; cf. T. Kohonen "Self-organised formation of topologically correct feature maps" in Biological Cybernetics, 43, 59 to 69, 1982. Here, an n-dimensional weighting vector, which is iteratively adapted by applying the method, is assigned to each neuron M on the SOM. Accordingly, each neuron M has a weighting vector w^M with n elements:

$$w^M = (w_1^M, w_2^M, w_3^M, \ldots, w_i^M, \ldots, w_n^M). \quad (6)$$

The number of elements or components of w therefore corresponds to the number of quantities n to be investigated with n = N + L in Equation (3). The number of neurons of an SOM is designated by k. An unfolded SOM network is generally shown as a rectangular map with $x \cdot y = k$ neurons; x designates the number of neurons in the x-direction and y the number in the y-direction. M is a selected neuron on the SOM with M = 1 ... k.

The weighting vectors w of all neurons k are distributed in the state space R by means of the algorithm. The SOM can organise itself on each n-dimensional structure (e.g. curve, surface or body), but can only implement an applicable and reproducible analysis of the process data by means of a modification of the method in accordance with the invention. Namely, two selected opposing corner neurons on the rectangular SOM map M_1 and M_2 are set to the minimum and maximum respectively of all process quantities in P and mathematically fixed there by making the weighting vectors of these corner neurons w^{M1} and w^{M2} equal to the vectors for

the scaled minima and maxima of the process quantities. Minimum therefore means the closest to the origin of the coordinates in R and maximum the furthest from this. As a result of this, the map will unfold in a reproducible manner, as a rotation of the map in R can be avoided.

The situation in the state space R is visualised using the U-matrix method or a dynamic visualisation method.

With the U-matrix method, the differences between the weighting vectors of each neuron with respect to its neighbour are calculated and displayed, appropriately graphically formatted, e.g. colour-coded. Connected areas (e.g. light surfaces) have a small difference, as their neurons have been placed in close proximity to one another in the state space; between these individual clusters there are boundaries, which are characterised by a high difference between the respective weighting vectors; these are visualised as dark surfaces, for example. In this way, a good visualisation of all cluster boundaries and thus the real process states can be achieved, see also G. Whittington, C. Spracklen: The Application of Neural Network Mode to Sensor Data Fusion in Proc. of Applications of ANN Conference, Orlando, USA, 1990.

In the dynamic methods, the input vectors are again provided to the unfolded map in random order and the current winning neuron determined in accordance with a "winner-takes-all" algorithm. This algorithm states that the neuron, whose weighting vector is closest to the input vector, wins. The statement "closest" is de-

termined by calculating a previously specified distance, e.g. the euclidic distance. The "winner-takes-all" algorithm is a method in which that neuron is always selected, which best fulfils a certain condition; all other neurons of the network or of the map are inactive or not selected (1-from-k selection, where k is equal to the number of neurons on the map). Here, this particular condition is a minimum distance between the current input vector and the weighting vector of the individual neurons, see S. Hafner, "Neuronale Netze in der Automatisierungstechnik", Oldenbourg Verlag, 1994, in particular Pages 17 to 25.

An internal counter, the winner rate, is increased by the numerical value 1 for the appropriate winning neuron. At the end of this dynamic visualisation, the winner rates for the individual neurons are recoded into colour or brightness values, a higher numerical value meaning a light colour, for example. The neuron with the highest winner rate is shown in white. Neurons with correspondingly lower rates are shown darker, for example. Neurons, which have not won at all, are shown in black on the SOM. As a result of this algorithm, light surfaces appear on the SOM, which represent the supporting points of the SOM, as the weighting vectors of these neurons lie very close to the data examples to be learnt. The map must interpolate between these supporting points. This visualisation method can therefore be used to build a security SOM. Values for the target quantities and the associated parameters on the light surfaces have a high reliability (security value near to 100%) and values on the intermediate dark surfaces have a lower reliability (security value considerably

less than 100%), as the map has interpolated there. A possible calculation for the security value SW is:

SW = current winner rate of the neurons/max. winner rate · 100% (7)

SW therefore specifies in percent how reliable the results determined by the map are. The maximum winner rate is the largest winner rate, which a neuron on the map has in comparison with all other neurons.

As a result of the above method, a structured SOM map is produced in which there is an association between the parameters p and the target quantities z. Here, an SOM neuron M, i.e. its weighting vector w^M, corresponds exactly to a point in R, i.e. one or more target quantities and the underlying influencing parameters. The positioning of the weighting vectors w in R for all neurons k of the map has been implemented by the self-organising learning process. At the same time, there are neurons, which represent the association between influencing and target quantities contained in the example data while other neurons specify unknown, non-linearly interpolated connections.

In the application phase of the map, the values of the target quantities or target quantity combinations that are of interest are sought out and the associated parameter combinations are determined by selecting the neurons on the SOM belonging to the target quantities. This choice and selection can take place graphically, as the target quantities can be plotted on the SOM in colour or as mountains.

A further description of the method is given below with reference to the exemplary embodiments shown in the figures of the drawing.

In the drawing:

Fig. 1 shows a structural diagram, which shows the main components of the system for carrying out the method,

Fig. 2 shows a possible weighting distribution of a neuron map in a process with only one influencing parameter and one target quantity (n = 2),

Fig. 3 shows a neural colour-coded security map for modelling showing supporting points (white surfaces on the map),

Fig. 4 shows a neural height-coded component map for a target quantity of the process with a path that has been drawn in for desired values of the target quantity,

Fig. 5 shows two neural height-coded component maps for two target quantities A and B of the process and a quality function C derived from this as a graphical, balanced superimposition of the two target quantities A and B for graphical polyoptimisation,

Fig. 6 shows a block diagram for combining the SOM with feed-forward networks for verifying the results.

In the following, the description of the method is divided into different method steps and explained by means of the above Figures by way of example.

Fig. 1 shows, by way of example, the structure of a system for carrying out the method for process analysis and diagnosis with a neural map.

Method step 1

Recording and selection of the process quantities

V1.1 Recording and formatting of the data. A data gathering and formatting system gathers and stores the process quantities required for optimisation.

V1.2 In a subsequent data analysis system, the selected and relevant data are subjected to a correlation analysis in order to determine the independent parameters. After this, the relevant quantities are scaled and, if necessary, noise is removed.

V1.3. Evaluation and analysis based on a neural system.

Method step 2

Map learning phase

V2.1. Unfolding of the self-organising neural network in the state space of the process based on modified algorithm according to T. Kohonen.

Fig. 2 shows, by way of example, the weighting distribution of SOM neurons in a 2-dimensional state space of the process with one influencing parameter (x-axis, $w(i, 1) = w1^i$) and one target quantity (y-axis, $(w(i,2) = w2^i)$. The consecutive number i (with i = 1...k) specifies the selected neurons i of the SOM network and thus the selected 2-dimensional weighting vector w^i. The position of the weighting vectors w of the neurons is marked by a grey circle and the neighbourhood relationships - of the SOM neurons belonging to the appropriate weights - are shown by a line. Neurons, which are connected by a line, lie adjacent to one another on the SOM map. The white circles mark points at which data vectors are present. It can be seen that the largest part of the SOM neurons has been placed here, however there are also neurons, which lie between the data vectors; these interpolate between these data examples.

V2.2. Visualisation of the unfolded network as a map by applying a dynamic visualisation method based on the appropriate winner rates of the neurons or by a statistical visualisation method according to the principle of the U-matrix method.

Fig. 3 shows - here in black and white - by way of example, the structured and colour-coded, neural security map for the manufacturing process of part of a machine. It represents the projection and visualisation of 30 process quantities on a neural map with 20 · 30 neurons. In Fig. 3, the x-axis specifies the number of neurons in the x-direction and the y-axis the number of neurons in the y-direction. In this example, the process quan-

tities consist of the target quantities torque and bearing temperature, and 28 influencing parameters.

The light surfaces are the determined support points; values of the target quantity torque, which are calculated at these points, are very reliable. The dark areas on the map form the envelope between these support points, as they show neurons, which are arranged in the state space between the learned data.

Although, in this example, each process vector is determined by 28 independent parameters (e.g. hole diameter, bearing length, material roughness) and 2 target quantities (torque and bearing temperature) and is therefore 30-dimensional, a topology-maintaining projection can be carried out by means of the method on only two dimensions, the SOM map. The number of simultaneously evaluated quantities is not restricted to 30; it can be considerably higher. The decisive factor is the topology-maintaining projection. In the above example, each target quantity represents a mountain on the 28-dimensional parameter space and, although this mountain exists, it is not accessible by conventional methods. The number of neurons does not depend a priori on the number of process quantities. The number of neurons should be chosen to be as large as possible; as a rule, it is restricted only by the computational power of the underlying neural system.

By means of the SOM map, the underlying topological connections of the quality functions in the high-dimensional state space can be maintained and can be projected and visualised on 2 or 3 dimensions. An ob-

server thus obtains an insight into the structure of the high-dimensional mountain of the quality function or - as in the example - of a target quantity.

As established, every neuron M on the map has a weighting vector w^M with n elements, with $w^M = (w_1^M, w_2^M, w_3^M, \ldots, w_i^M, \ldots, w_n^M)$. The i-th component of a weighting vector is designated by w_i. A component map is produced by visualising the values w_i^k of all neurons k of the map by means of the SOM. As the weighting values of each neuron can be unambiguously assigned to the value of the underlying process quantities, this representation produces a distribution of the respective target quantity j over the whole state space of the process. When converting from the weighted value to the real value of the target quantity, it may be necessary to take account of the scaling. In the case of the mountain, the values so obtained by means of the SOM are visualised with height coding. It is equally possible to convert the value of the target quantity into a colour value and to plot this on the SOM. The value of the target quantity can then be read off by means of an appropriate scale.

Fig. 4 shows by way of example the height-coded mountain of the target quantity torque projected on an SOM. This picture is produced by building up a height-coded component map for the torque component. Here, the x and y-axis of the map represent the definition range of the target quantity and the z-axis represents the range of values of the target quantity, i.e. the associated value of the target quantity is calculated for every SOM neuron and plotted in a 3-dimensional plot. It must

be noted that the definition range consists not - as is usually the case - of physical values of two influencing parameters, but of all 28 influencing quantities simultaneously. These 28 influencing parameters have been shown together with the target quantities on the map in such a way as to maintain the topology.

Method step 3

Application - optimisation of a target quantity of the process

As the real values of the target quantity can be plotted as mountains on the SOM, the required values can simply be marked on the mountains, and the associated parameter combinations can be output. Here, the target values can be plotted as contour lines, for example, on the mountains or it is possible to define a path P of permissible values on the target quantity mountains and to have all the associated parameter combinations for this path calculated from the SOM. Fig. 4 shows such a path - as a thick black line - on the mountains of a torque. The target value so defined lies at 700 when normalised. Furthermore, it is possible to select permissible ranges $\Delta z = z_{MAX} - z_{MIN}$ for the target value. In the height-coded representation, this range corresponds to a tolerance width in the Z-direction; this range is therefore to be looked upon as a double section parallel to the x,y plane, once in the height z_{MAX} and once for z_{MIN}.

The basic idea is that the underlying optimisation parameters can be determined starting from a required target value on the SOM.

The reverse calculation from the target value to the parameter settings is therefore possible because an SOM neuron can be assigned to each value of the target value in the definition range. For example, the neuron M on the map is chosen for the required value Z2 in Fig. 4. By this selection of the neuron M, the appropriate weighting vectors w^M are output and converted into the real values of the process parameters. By this means, the sought-after parameter configurations, p_1, p_2, ..., p_N, which have just led to the required target value, can be determined exactly. These parameter combinations that have been found can be checked for plausibility by means of a subsequent procedure and, if required, output to the process on-line or off-line.

As the target value is graphically visualised by the mountains, it is also possible to select the minima or maxima of the target value without any difficulty; further, the embedding of this value in the context of other values can also be seen for the case where there are several ranges with the same values of the target quantity. Robust and error-tolerant optimisations can therefore be determined, as, in case of doubt, the target value on a steep slope of the mountain, e.g. Z1 in Fig. 4, would not be chosen if there were still an alternative on a plateau, e.g. Z2 in Fig. 4. Both points correspond to an equal numerical value of 800 for the target quantity, see Z-axis, but Z2 is considerably more stable. By this means, parameter combinations,

which in addition are still relatively robust with respect to small changes in the parameter values, are found for a required target quantity.

Accordingly, the actual optimisation is carried out by selecting the required target values z on the SOM mountain and subsequently calculating the underlying parameter combinations $p_1, \ldots, p_N$ by means of the SOM map. The SOM map can therefore be used as an inverse model f with $(p_1, \ldots, p_N) = f(z_j)$.

Furthermore, the selection of the required target value is made in the context of the whole of the target quantity mountains. In this way, for the case where there are several similar target values, those values can be found, which are most suitable for the optimisation task with regard to other characteristics. Most suitable could, for example, be the most robust, as the target quantity lies on a plateau of the mountains.

The search for minima or maxima is also very easy, as the position of a mountain MAX or a valley MIN can be seen from the mountains without any difficulty. A minimum or a maximum is found exactly at these points and the associated parameter combination can be output. The decisive factor is that the map can interpolate and that therefore target values, which were not present in the optimisation examples, can be displayed and selected. By this means, in conjunction with the context-dependent selection, it is possible to implement robust optimisations, which can be carried out without an underlying mathematical model and which are therefore universally applicable.

Furthermore, if the results of the security SOM shown in Fig. 3 are used for the reliability of the SOM calculation, then it is also possible to discriminate between reliable and unreliable optimisation results on the mountains. A conclusion can therefore be drawn as to how probable it is that the characteristics of the SOM mountain correspond with the reality in the state space P. Light areas of the mountain have a high level of reliability; dark depicted areas correspond to uncertain mountain characteristics.

The plateaus of the target quantities are particularly interesting when robust optimisations are sought, i.e. optimisations in which changes in the influencing parameters do not lead to serious changes of the target quantity. Particularly for this so-called error-tolerant optimisation, these plateaus are a completely new way of selecting parameters. It is generally applicable, non-linear and can be intuitively understood, as a target value on a plateau means that, for smaller changes of values of the influencing quantities, this will move about on the plateau and that, as a result, its numerical value will only change very slightly. On the other hand, the value of a target quantity (e.g. z_M, whose associated neuron M lies on the slope of a mountain, will respond very sensitively to changes in the influencing quantities $p_1, \ldots, p_N$. Here, too, small changes of $p_1, \ldots, p_N$ only have the effect that neurons around the original winner neuron M are selected; the values of the target quantity for these adjacent neurons can change greatly, however, as the value of their target quantity lies on a slope of the mountain.

Furthermore, it is possible to calculate a quality function Q based on the values of a target quantity and to display this instead of the target quantity itself.

Method step 4

Application - Polyoptimisation of several target quantities

In the case where several target quantities L have been trained with the map ($2 <= L << N$), these target quantities can be evaluated together by introducing a quality function G, which is calculated from several target quantities. As each target quantity can be represented as mountains on the map, a completely new set of mountains with modified characteristics is obtained by superimposing these mountains. The minima and maxima on these new mountains show that here, for example, the target quantity vector of all target quantities is minimum or maximum. By this means, the best compromises for the superimposed target quantities can be found.

Fig. 5 shows an example in which the target quantity torque (Part A, Fig. 5) and bearing temperature (Part B, Fig. 5) have been superimposed. The resulting mountains GEB are shown in Part C of Fig. 5. There are several possibilities for this superimposition: on the one hand, a simple additive superimposition of the target quantities can be implemented by adding the L target quantity values, which are stored in the weighting vector w of a neuron M, for each neuron M on the map:

$$value_G^M = value_{z1}^M + value_{z2}^M + \ldots + value_{zj}^M + \ldots + value_{zL}^M \quad (8)$$

with M = 1 ... k and $value_G^M$ as the value of the new quality function G - calculated from the vector - as the starting value for the j-th target quantity at position M on the map. The result of a simple additive superimposed addition of the two target quantities torque and bearing temperature is shown in Part C of Fig. 5.

On the other hand, it is also possible to weight certain target quantities more than others. This is done by carrying out a weighted superimposed addition of the target quantities z_1, z_2, .., z_L.

$$value_G^M = a_1 \cdot value_{z1}^M + a_2 \cdot value_{z2}^M + \ldots + a_L \cdot value_{zL}^M \quad (9)$$

In general, this allows any required quality function to be calculated from the target quantities with

$$G = f(z_1, z_2, \ldots, z_L). \quad (10)$$

The function f can be any mathematical function. After calculating the new values $value_G$ at all points on the map, i.e. for all neurons, and, if necessary, after subsequent scaling, the result is plotted once more as mountains GEB on the map. In this way, a universal optimisation can be carried out for several target quantities simultaneously, as the underlying parameter combinations can be calculated for each required target value according to (p1 ... pN) = $g(f(z_1, z_2, \ldots, z_L))$

where g corresponds to an inverse mapping between the parameters P and the target quantities Z. The function g does not have to exist in the mathematical sense; it was determined by the SOM method during unfolding.

As the quality function G can also always be represented as mountains GEB, the method constitutes an extension of current methods for multi-quantity optimisation (poly-optimum) in research, development and design and in industrial and chemical processes. The advantage is that a minimum or maximum, also of the compromise solution, can easily be determined graphically and the associated parameter combinations can be output in each case. The user only has to link together the target quantities of his choice and the resulting quality function will be displayed as new mountains.

Method step 5

Application - verification of the results of the analysis

For verifying the parameter optimisation, a neural feed-forward network is used, which has modelled the same test process as the SOM. In contrast to the SOM, the inputs and outputs are treated separately in the neural feed-forward networks. All parameters, i.e. all influencing quantities for the optimisation, are applied to the network inputs. All target quantities are connected to the network outputs. By using different neural learning methods, the network learns an approximation between its inputs and outputs. The whole process can be modelled by this means.

A feed-forward network is a term introduced for a class of networks, the neurons of which are arranged in layers and in which there are no feedbacks between neurons of a higher layer to neurons of a lower layer; see A. Zell, "Simulation Neuronaler Netze", Addison Wesley Verlag, 1st edition, 1994, Pages 76 to 78. These networks are suitable for building up static and dynamic models between selected quantities. Starting from a neural model, which has been built up between the test parameters p_1, p_2, ... p_N and the target quantity z_j, this model can be used for verifying the parameter optimisation.

Fig. 6 clarifies the principle. Here, the feed-forward network model must be trained with the same test examples as the SOM. A security - RBF - model is built up as a neural feed-forward network model. A security network is a network that, in addition to its model output value, specifies how reliable this model output value is. This reliability can, for example, be output by means of a numerical value of 0 ... 100%. Here 0% means no reliability, 100% means that the network result is completely reliable.

It is possible to build up such security networks when using suitable local approximating networks, e.g. RBF networks (see A. Zell, Pages 225 to 239). RBF networks are 3-layer networks, the hidden layer of which consists of neurons, which have a Gaussian activation function. Their maximum output activity is limited to a small input range. These neurons are therefore only sensitive for a small, n-dimensional, hyperspherical

input range. If the input vector lies in this range, the respective hidden neuron has its maximum output activity; the other hidden neurons have a correspondingly lower activity (see A. Zell).

By calculating the magnitude of a distance between the applied input vector and the hidden neuron with maximum activity, a measure is obtained of whether the network is currently being used in a range in which it has not been trained. If the magnitude of this distance is large, this means that the input vector is a very long way from the supporting points of the RBF network. The result of the network is therefore relatively unreliable, as neural networks interpolate between the learned supporting points. This interpolation is even more unreliable the further the value is from the nearest supporting points. If the current output value is in the extrapolation space, the result is even more unreliable.

An example for calculating the security value SW_{RBF} is:

$$SW_{RBF}[\%] = activity_{WINNER}[\%] - MSE_{Learn}[\%] \quad (11)$$

where SW_{RBF} is the reliability value of the network response in percent, $activity_{WINNER}$ specifies the activity response of the winner neuron to an input value multiplied by 100%, and MSE_{Learn} is the mean learning error of the network, recorded during the training. The security value can therefore not be greater than the accuracy during the learning process. The activity value is a measure of the distance from the input vector to the next learned supporting point. This results in a secu-

rity RBF network, which calculates and outputs not only the result of the simulation but also the reliability of the result.

V5.1. The optimum test parameters $p_{1(opt)}, \ldots, p_{N(opt)}$ determined for a selected target quantity value z_j in Fig. 6 are fed to the security RBF network and the associated target quantity $z_{j,TEST}$ is determined. If $z_{j,TEST}$ corresponds approximately to z_j, i.e. if the difference DIFF = $|z_j - z_{j,TEST}|$ is less than a pre-defined value OK, then the results of the SOM are verified, as the two underlying different neural models have calculated the same model results. N verified, optimum parameters $p_{1(opt)}, \ldots, p_{N(opt)}$ have therefore been confirmed for the required target value. In the negative case, the selection $z_{j,TEST}$ must be repeated iteratively and the SOM or RBF model retrained.

This means that, starting from the required target value z_j, the associated parameters are sought with the SOM and these results are verified with the security RBF network, as the respective parameters must again lead to the target value selected on the SOM.

V5.2. In the case of polyoptimisation, the parameter combinations sought with the SOM for a compromise value (e.g. minimum or maximum) by means of the security RBF network must lead approximately to the individual values of each target quantity selected on the SOM.

Present optimisation methods can be extended by the combination of the target-orientated approach of the SOM with the verification possibilities of a feed-

forward model, as the methods described here can be applied non-linearly and universally, i.e. without an underlying mathematical model. In particular, optimisations can be determined, which are robust with respect to changes in the optimisation parameters p.

A further advantage of the method lies in the low engineering outlay. As the learning and structuring of the maps is produced completely by the method, extensive engineering or parameterisation of rules, models or differential equations can be dispensed with. The engineering is an integral part of the method itself.

Patent claims

1. Method for solving optimisation tasks in research, development and design and for the optimisation of technical or chemical products and of processes for manufacturing these products, in which all the quantities relevant to the optimisation are brought together and evaluated in relation to one another by a neural analysis based on self-organising maps, so-called SOMs, in which a topology-maintaining, non-linear projection of data of the relevant parameters and the associated target quantities is implemented on a multi-dimensional SOM, this being visualised as an SOM component map for optimising a target quantity either height-coded in the form of mountains (GEB) or colour-coded, and after selecting values for the target quantity (Z1, Z2) on the SOM component map, the underlying parameter combinations are calculated and output.

2. Method according to Claim 1, characterised in that products from mechanical and automotive engineering, system building, precision engineering and the chemical and genetic engineering industry are optimised as well as materials.

3. Method according to Claim 2, characterised in that products from system building are power station engineering components.

4. Method according to one of the preceding claims, characterised in that, in the case of a height-coded representation, the required values of a target quantity are selected by one or more contour lines on the

target quantity mountains or by defining a path (P), a range or a plateau on the mountains.

5. Method according to one of the preceding claims, characterised in that the target quantity mountains are compared with the reliability values of a security SOM in order to identify the ranges on the SOM, which represent the most reliable values for the optimisation.

6. Method according to one of the preceding claims, characterised in that several target quantities are optimised simultaneously and in conjunction with one another in the manner of a polyoptimisation by linking the selected target quantities with one another by means of any mathematical function and plotting the resulting quality function (G) as mountains (GEB) on the SOM and determining the underlying parameter values starting from the new mountains.

7. Method according to Claim 4 or 6, characterised in that contour lines, paths, ranges or plateaus are entered on the superimposed mountains (GEB) and the associated parameter combinations are determined.

8. Method according to one of the preceding claims, characterised in that it is carried out in the following steps:

a) Recording and selection of the test parameters and formatting of the data,

b) Unfolding of the self-organising neural network in the state space of the process based on the SOM algorithm, wherein the values of the relevant test parameters are used,

c) Representation of the unfolded network as a neural map,

d) If necessary, projection of the high-dimensional parameter space on the neural map based on the U-matrix method or by means of so-called winner-takes-all algorithms with subsequent visualisation of the summed winner rates of the individual neurons on the map for building up a security SOM,

e) Representation of the target quantities height-coded or colour-coded on the SOM,

f) In the case of a height-coded representation: determination of contour lines, paths or ranges on the SOM for selecting the required values of the target quantities,

g) Calculation of the underlying parameter combinations for the required target values,

h) If necessary, weighted or unweighted superimposition of the mountains of several target quantities and definition of the required ranges on the newly resulting target quantity distribution or quality function distribution.

9. Method according to one of the preceding claims, characterised in that the results, which were obtained by means of the SOM, are verified by a subsequent security RBF model.

In addition 5 page(s) of drawings

Abstract

Method for optimising products and production processes

The invention relates to a method for solving general optimisation tasks in research, development and design and for the optimisation of technical or chemical products and of processes for manufacturing these products. In order to enable a simultaneous and consistent evaluation of relevant parameters and the associated target quantities, it is proposed that the effects of the process quantities be evaluated in relation to one another by a neural analysis based on self-organising maps, in which a topology-maintaining, non-linear projection of data of the test parameters is implemented on a 2-dimensional neural map (so-called SOM). After selecting values for the target quantity (Z1, Z2) on the SOM component map, the underlying parameter combinations are calculated and output.

Key to Figures

Figure 1

Zielgrößenauswahl	Target quantity selection
Optimierungssystem / SOM	Optimisation system / SOM
Optimierer	Optimiser
Neuro-System	Neural system
Prozeßparameter und Zielgrößen	Process parameters and target quantities
Datenanalyse-System (Auswahl und Korrelator)	Data analysis system (selection and correlator)
Datenaufbereitungs-System / Daten-Server	Data formatting system / Data server
Prozeß / Datenbank	Process / Database

Figure 4

Neronale Komponenten-Karte für Zielgröße Drehmoment	Neural component map for target quantity torque